



A GPU Algorithm for Solving the Positions of New Pulsars

Ziyao Fang (方子瑶)^{1,2}, Weiwei Zhu (朱炜玮)² , Chenchen Miao (缪晨晨)³, Yukai Zhou (周宇凯)^{2,4}, Dejiang Zhou (周德江)² , Tianlu Chen (陈天禄)¹ , Qiuyang Fu (付秋阳)^{2,5}, Lingqi Meng (孟令祺)^{2,5}, Xueli Miao (缪雪丽)², Jiarui Niu (牛佳瑞)^{2,5} , and

Mengyao Xue (薛梦瑶)²

¹ The Key Laboratory of Cosmic Rays (Tibet University), Ministry of Education, Lhasa 850000, China

² National Astronomical Observatories, Chinese Academy of Sciences, Beijing 100101, China; zhuww@nao.cas.cn

³ Research Center for Intelligent Computing Platforms, Zhejiang Laboratory, Hangzhou 311100, China

⁴ School of Physics, Peking University, Beijing 100871, China

⁵ School of Astronomy and Space Science, University of Chinese Academy of Sciences, Beijing 100049, China

Received 2024 April 2; revised 2024 July 18; accepted 2024 July 30; published 2024 October 25

Abstract

Timing newly discovered pulsars requires gradually building up a timing model that connects observations taken days to months apart. This sometimes can be challenging when our initial knowledge of the pulsar's position is arcminutes off from its true position. Such a position error leads to significant arrival time shifts as a result of the Earth's orbital motion. Traditional down-hill fitting timing algorithms become ineffective when our model predicts the wrong pulse rotations for our next observation. For some pulsars whose model prediction is not too far off, the correct rotation number could be found by trial-and-error methods. For the remaining challenging pulsars, a more generalized method is called for. This paper proposes a GPU-based algorithm that could exhaustively search a large area of trail positions for probable timing solutions. This could help find phase-connected timing solutions for new pulsars using brute force.

Key words: methods: data analysis – (stars:) pulsars: general – (stars:) pulsars: individual (J0706+2707, J2149+5643)

1. Introduction

Pulsars are powerful tools for studying fundamental physics, gravitational theories, and gravitational waves (Wex 2014; Berti et al. 2015; Kramer et al. 2021; Paulo 2024). Measuring the mass of neutron stars could help us understand the equation of the state of super-nuclear matter and the mechanism of neutron star formation (Antoniadis et al. 2024; Fonseca et al. 2016). Pulsars could probe the Milky Way's large-scale structures in magnetic fields and ionized medium (Alvarez-Muñiz & Stanev 2006). The extreme density, pressure, and magnetic field of neutron stars, as well as their internal super-fluidity and super-conductivity, provide the ideal labs for testing the mechanism of particle acceleration in the magnetosphere, high-energy radiation, electromagnetism, and relativistic plasma physics (Ruderman & Sutherland 1975). Furthermore, the pulsar timing array has detected evidence for nano-hertz gravitational waves (Antoniadis et al. 2022; EPTA Collaboration et al. 2023; Reardon et al. 2023; Xu et al. 2023).

The aforementioned scientific cases start from discovering pulsars in surveys and solving and timing the discovered pulsars. More than 3534 pulsars have been discovered until 2024 January, according to the ATNF catalog version 2.1.1⁶ (Manchester et al. 2005); among them, 574 are so-called

millisecond pulsars, i.e., pulsars with a spin period of fewer than 30 ms.

Solving new pulsars means determining their nature through a series of initial timing observations. This is done by modeling the time of arrival (TOA) with a series of pulse numbers and phase values as predicted by a timing model. The modeling usually could be well constrained in several observations taken over a few years for an isolated normal pulsar. However, a millisecond pulsar in a binary orbit often requires more effort because a binary system contains more parameters that must be determined well to phase connect all the observations. More importantly, a millisecond pulsar has a small rotational period, which could be much smaller than the uncertainty of model prediction when projecting months into the future. Without predicting the correct rotation number, traditional down-hill fitting algorithms cease to work in improving the timing model. In this case, one usually has to conduct more observations in a short span of time, trying to keep more TOA phases connected and improve the precision of the model so that there is no more pulse number ambiguity. This process sometimes could take a long time and many extra observations, even though there are already more than enough TOAs to provide strong constraints to the timing model.

Freire & Ridolfi (2018) developed the Dracula program script that automatically tested a number of possible pulse numbers for each epoch of TOA and tried to find the correct

⁶ www.atnf.csiro.au/people/pulsar/psrcat/

timing solution through trial and error. Phillips & Ransom (2022) developed APT for solving isolated pulsars, and Taylor et al. (2024) developed APTB for solving binary pulsars following trial and error methods. This approach could help solve some millisecond pulsars but would take too long to run when the real pulse number solution is too many rotations away from the current best guesses. In these difficult cases, the offset in pulse numbers is often caused by the pulsar's positional errors.

This paper proposes a GPU-based algorithm that could exhaustively search over a range of R.A. (α) and decl. (δ) values with necessary fine grinding to brute-force solve a pulsar. The algorithm could help solve millisecond binary pulsars when the binary model is reasonably well constrained, and R.A., decl. errors were the main cause of TOA offsets. The search in position space requires fine gridding. In our test cases, we must search over square arc minutes of parameter space with a milliarcsecond grid, i.e., mounting to a 10^9 – 10^{10} position trials. Luckily, this computation is highly parallelizable, making it ideal for a GPU. Our program is a complement to the existing pulsar-solving tools.⁷

2. Method

When a radio pulsar is discovered, we first learn its dispersion measure (DM), spin period, and rough position, which is precise to a few arc minutes. We then conduct a series of observations to determine its nature. If the pulsar's spin period is stable in the following observations, it is likely an isolated pulsar; otherwise, it could be a binary system.

Several scattered observations over a year often suffice for an isolated pulsar to determine its precise spin parameters and position. For a binary system, dense observations are often needed to identify the orbital period. In these close observations, we observe how the pulsar's spin period varies over time and infer the binary system's initial parameters by fitting the spin periods and sometimes the spin period derivatives by using a customized program or existing program like PRESTO⁸ (Ransom 2011).

The initial spin parameters (often only the spin period) and the initial binary parameters constitute the binary pulsar's initial solution. They could be further refined by fitting the pulsar's TOAs in timing software such as TEMPO (Nice et al. 2015), TEMPO2 (Edwards et al. 2006), and PINT (Luo et al. 2021).

For most pulsars, we could continue refining our initial solution by extending our observations once per month and adding fit to position and spin period derivatives without losing phase connection. Eventually, we could get a long-term stable timing solution that reflects the pulsar's real position and spin

properties. But for a few exceptional pulsars, often millisecond pulsars, we might find it hard to keep the phase connection after several months when the extra residuals caused by the pulsar position error become significantly larger than the pulsar period. This is because to continue our timing practice, we need our best model to predict the next epoch's pulse number so that the TOAs of the last epoch would be placed within one rotation of the correct model. If our model is somehow inaccurate enough to predict the wrong rotation number for our next epoch, we often cannot immediately get a new solution that could work with the new TOAs. This problem is called losing phase connection. This problem could sometimes be solved by adding or subtracting a few rotations or subtracting from our prediction, but sometimes, it requires extensive experimenting. For this reason, Freire & Ridolfi (2018) program the DRACULA script that automatically experiments by adding or subtracting a few rotations from observation epochs disconnected in phase from earlier ones. This is an effective way to help solve those challenging pulsars, but not a method guaranteed to work out. Some pulsars require more experiments than the loop-based script. The model predictions could be off by more than dozens of rotations on several different epochs, making it difficult for trial-and-error methods to reach them. An effective method is needed to exhaustively search for the correct solution under these circumstances.

So, we develop a GPU-based code to find the correct solution by searching the most probable range of R.A. and decl. A fine grid is needed in this search because the effect of position error could be significant on the TOAs. Let us assume a position error of $\Delta\alpha \sim 1''$ (~ 0.0003 rad); the TOA error that this introduced could be roughly estimated by 0.0003×500 s, i.e., ~ 0.15 s. For a millisecond pulsar with a spin period of 1.5 ms, the position-induced error could be nearly 100 rotations. To search and place our solution within one rotation of the correct model, we need to search for position space in $0.''1$ precision. But our algorithm needed to search and place the solution to nearly the same level as TOA errors, which are often at $10 \mu\text{s}$ level; this requires us to search in position space with a $1 \sim 10$ mas precision.

The pulsar rotation model is usually expressed in the reference frame co-moving with the pulsar. Since pulsars do not rotate at a constant pulse frequency, we typically use a Taylor expansion to describe the rotational phase as

$$N(t_{\text{PSR}}) = N_0 + \nu_0(t_{\text{PSR}} - t_0) + \frac{1}{2}\dot{\nu}_0(t_{\text{PSR}} - t_0)^2 + \frac{1}{6}\ddot{\nu}_0(t_{\text{PSR}} - t_0)^3 + \dots, \quad (1)$$

here $N(t_{\text{PSR}})$ is the phase/pulse number of the TOA t_{PSR} observed in the pulsar's co-moving reference frame, N_0 is the phase/pulse number at a reference epoch t_0 , ν_0 is the pulse frequency at t_0 , $\dot{\nu}_0$ and $\ddot{\nu}_0$ are the first and second derivatives of pulse frequency.

⁷ <https://github.com/ZiyaoFang/DARG>

⁸ <https://github.com/scottransom/presto>

To compute a pulsar's timing residual, we need to perform a time transformation from a topocentric TOA (t_{topo}) to the pulsar time (t_{PSR}):

$$\begin{aligned} t_{\text{SSB}} &= t_{\text{topo}} + t_{\text{corr}} + \Delta_{R_{\odot}} + \Delta_{S_{\odot}} + \Delta_{E_{\odot}}, \\ t'_{\text{PSB}} &= t_{\text{SSB}} - \frac{\Delta D}{f^2} - d/c, \\ t'_{\text{PSR}} &= t'_{\text{PSB}} + \Delta_B(x, P_B, T_0, e, \omega). \end{aligned} \quad (2)$$

On the left-hand side of Equation (2), SSB represents the center of mass of the solar system, where t_{SSB} is the TOA at the SSB reference frame. t'_{PSB} is the theoretical pulse time at the pulsar binary system barycenter expressed in the same coordinate as the solar system barycenter time, and t'_{PSR} is the pulsar's reference frame pulse time (t_{PSR}) expressed in the same coordinate as the solar system barycenter time. t'_{PSR} is different from t_{PSR} by an unknown Doppler factor. We usually ignore this unknown Doppler factor and use t'_{PSR} to compute the time and phase residual of the pulsar as described in Equation (1).

On the right-hand side of Equation (2), t_{topo} is the topocentric arrival times observed with the observatory clock, and t_{corr} is the clock corrections to the topocentric TOAs. $\Delta D/f^2$ is the time delay of pulses due to dispersion in the interstellar medium, where $\Delta D = D \times \text{DM}$, D is a dispersion constant when the DM is expressed in the units of $\text{cm}^{-3} \text{pc}$ and frequency f is in MHz. d is the distance to the pulsar, and c is the speed of light in the vacuum. Since d/c is regarded as a constant, it can be omitted from our calculation. $\Delta_{R_{\odot}}$ is the solar system Römer Delay, a time delay from the light-travel between the phase center of the telescope and the SSB (Lorimer & Kramer 2005). The Römer Delay varies significantly with the pulsar's position so it is the main effect we consider (3). $\Delta_{S_{\odot}}$ is the solar system Shapiro Delay due to the gravitational perturbation of the light-path (Shapiro 1964) from all solar system bodies, generally in the initial solving process, we only consider the delays caused by the Sun and major planets (such as Jupiter). $\Delta_{E_{\odot}}$ is the solar system Einstein Delay comprised of gravitational redshift and time dilation (Taylor & Weisberg 1989). Δ_B is the time delays from the pulsar's binary system, such as the Römer Delay, which is a function of the pulsar's orbital parameters: projected semimajor axis x , orbital period P_b , eccentricity e , periastron time T_0 , and the angle between periastron and the ascending node ω . Because Δ_B does not rely on the pulsar's position, we do not describe it in detail here. Once we have deducted all the solar system, propagation, and pulsar binary effects, the resulting t_{PSR} only differs from the emission time of the pulsar's pulse by an unknown constant Doppler factor. Therefore, this t_{PSR} could be modeled by the simple rotational phase model mentioned in Equation (1).

Among all the solar system effects, only the Römer Delay varies significantly with pulsar position, so the TOA offset Δt_{topo} due to position error ($\Delta\alpha$ and $\Delta\delta$) can be expressed

approximately as:

$$\begin{aligned} \Delta t_{\text{topo}}(\Delta\alpha, \Delta\delta) &\approx \Delta t_{R_{\odot}}(\Delta\alpha, \Delta\delta) \\ &= -\frac{(\hat{s}_{\text{new}} - \hat{s}_{\text{initial}}) \cdot \mathbf{r}_{\text{earth}}}{c} \\ &= -\frac{\mathbf{r}_{\text{earth}}}{c} \cdot \begin{bmatrix} \cos \alpha \sin \delta \cdot \Delta\delta + \sin \alpha \cos \delta \cdot \Delta\alpha \\ \sin \alpha \cos \delta \cdot \Delta\delta + \cos \alpha \cos \delta \cdot \Delta\alpha \\ \cos \delta \cdot \Delta\delta \end{bmatrix}. \end{aligned} \quad (3)$$

Here, \hat{s}_{initial} , \hat{s}_{new} are the initial and new values of unit vectors pointing from the Solar system barycenter to the barycenter of the pulsar system; $\mathbf{r}_{\text{earth}}$ is the vector pointing from the SSB to the phase center of the telescope, and c is the speed of light. Our program uses the GPU to conduct a grid search over $\Delta\alpha$ and $\Delta\delta$ using a fine grid of the order of milliarcseconds to find our desired solution.

Therefore, suppose we move to a new position, $\Delta\alpha$, $\Delta\delta$ away from our initial position; we will get a new set of TOA residuals R_{new} that is offset from our initial residuals R_{initial} :

$$R_{\text{new}} = R_{\text{initial}} + \Delta t_{\text{topo}}(\Delta\alpha, \Delta\delta). \quad (4)$$

If the new position is exactly the correct position of the pulsar, the new residuals will likely form a linear trend or a linear trend with some TOAs offset from the trend by an integral number of rotations. Here, we assume that the trend would be linear because only the error of the spin period manifests into the residual, and the spin derivative is not yet a prominent effect. If the new residuals form a linear trend, we could solve for the trend's slope k and cutoff b using the least-square method:

$$\begin{aligned} k &= \frac{\sum_{i=1}^{N_{\text{toa}}} (x_i - \bar{x}) \cdot \sum_{i=1}^{N_{\text{toa}}} (y_i - \bar{y})}{\sum_{i=1}^{N_{\text{toa}}} (x_i - \bar{x})^2} \\ b &= \bar{y} - k \cdot \bar{x} \end{aligned} \quad (5)$$

where x_i , y_i is TOA (t_{topo}) and new residual R_{new} of each pulse, and \bar{x} , \bar{y} is the mean value of TOAs and new residuals. We can revise the offset of the spin period according to the least χ^2 solution with

$$P_{\text{new}} = P_{\text{initial}} \cdot (1 + k/(24 \times 3600)) \quad (6)$$

where k is the trend's slope, P_{initial} is the spin period of the initial model, and P_{new} is the new period updated according to the new position.

We show the steps in solving a normal pulsar J0706+2707 in Figure 1. First, we use an initial model that contains only spin frequency (ν_0) and the initial orbital parameters (x , P_b , T_0 , etc...), omitting $\dot{\nu}_0$ and try to solve the pulsar step by step, until we cannot connect the last TOA. The resulting residuals are shown in the top panel. Second, we search for possible positions in the error box of the initial observed position. For this normal pulsar, the spin period is ~ 70.3 ms, which is larger than the residual offset due to the position offset in most cases. When we grid search and find the correct position, we can

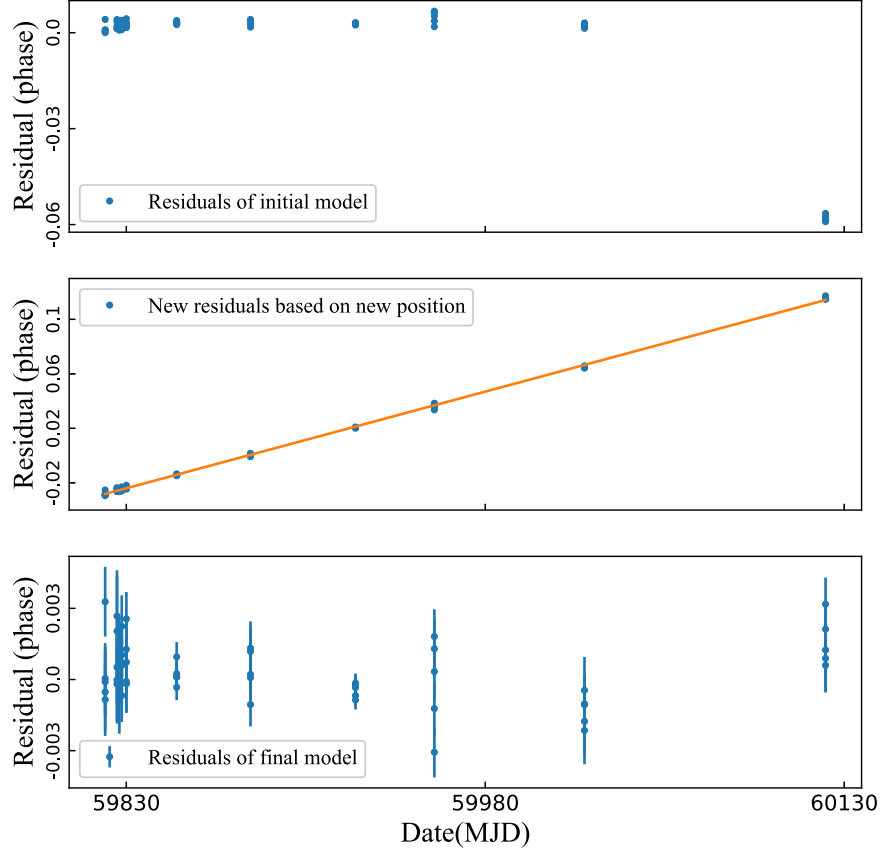


Figure 1. The steps in solving the normal pulsar J0706+2707. Top panel: The timing residuals of the initial model with the spin period of 70.3 ms. Middle panel: The residuals of the calculated model, which replace the original model’s position parameters R.A. and decl. with the correct position; Bottom panel: Timing residuals of the new model with updated R.A., decl., and spin period.

obtain new residuals that form a linear structure as shown in the middle panel of Figure 1. Finally, we use the least-square fit of the linear trend to update the pulsar’s period parameter and recalculate the residual in the GPU for the new position, the resulting residual is shown in the bottom panel of Figure 1.

Before we begin solving a binary pulsar’s position, we have to first estimate the binary system’s orbital parameters based on the pulsar’s first few observations. We fit the pulsar period measured in the first several observations with a circular orbit model using `fit_circular_orbit.py` (Ransom 2011) to estimate the Keplerian parameters. e.g., orbital period (P_b), the epoch of periastron passage (T_0), and the projected semimajor axis (x). Then we fit the TOAs of those first observations in TEMPO2 using the estimated Keplerian parameters (P_b , x , and T_0) in an ELL1 timing model (Lange et al. 2001) while fixing the pulsar position parameters to our discovery position. This procedure allows us to derive an initial timing solution, including a binary model that works well until the errors in our pulsar position start to cause significant residuals. One should note that such a method likely only works in pulsar-white dwarf systems with very low eccentricities. For systems with non-negligible

eccentricities, one has to use a more comprehensive method such as a non-circular orbit fitter. Taking PSR J2149+5643 as an example, its $x \sim 8.2$ ls, and $e \sim 10^{-5}$, this means that the maximum residual offset caused by not including the eccentricity is about $ex/c \sim 0.1$ ms, this corresponds to ~ 0.007 in its rotation phase. This is a relatively small offset that will not significantly affect our procedure and can be corrected in the end by adding a few more timing parameters to our model.

Initial timing solution obtained this way could often make the TOAs phase-connected in the earliest several months but then stop connecting after the residuals caused by position error become significant. After that, we start fitting or searching for a better pulsar position using our GPU program.

For millisecond pulsars in a binary system, the spin period is often far smaller than the residuals caused by position errors. In this case, the rotation number of the TOAs, especially the last one that is not connected, could be offset from the true value. The steps for solving this pulsar using GPU are shown in Figure 2.

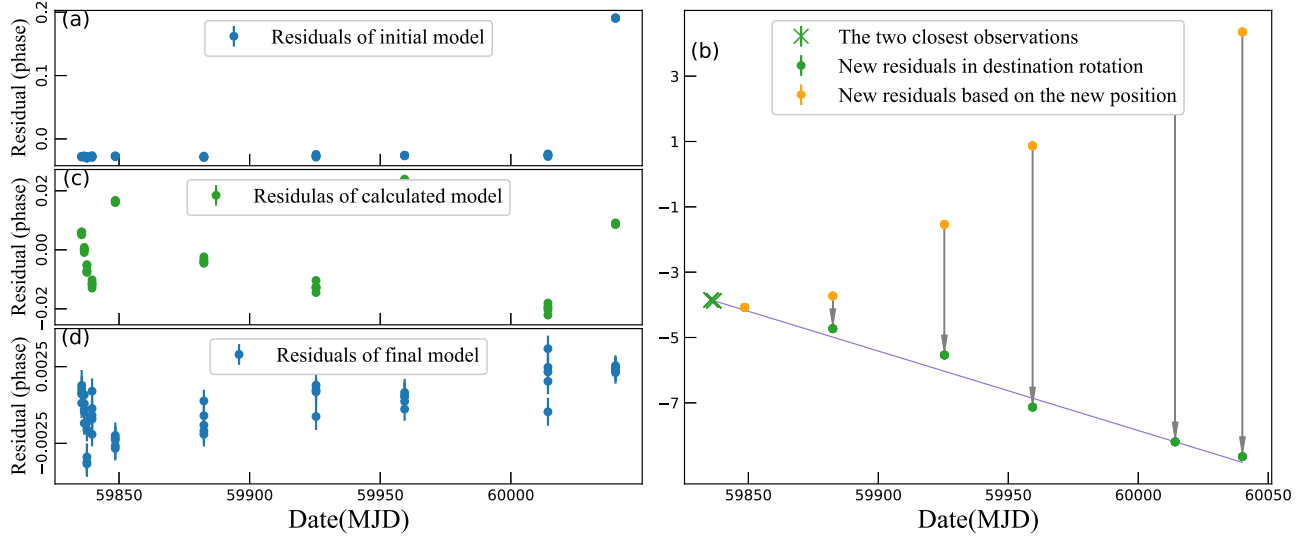


Figure 2. The steps for solving the millisecond pulsar J2149+5643. Left top panel (a): The residuals of the initial model from a millisecond pulsar with a spin period of 15.3 ms. Right panel (b): Calculate the new residuals based on the new position and move to the predicted rotation number. Left middle panel (c): The timing residuals with updated position and spin period. Left bottom panel (d): The timing residuals with updated position, spin period, and orbital parameters.

In the first step, we calculate the pre-fit residuals of the initial model, which are shown in panel (a) of Figure 2. As one can see the last TOA residuals are already deviating from 0 by a significant fraction of phase. What is not shown in this plot is the phase residual of the TOAs with respect to the pulse numbers from panel (a), the last one of which could be many rotations away from the true value. Now, we need to determine the correct pulse number to which R_{new} belongs to ensure the continuity of all TOA phases. So in the second step, we update the pulsar position in the timing model and use GPU to compute the residuals for the updated model, this is shown as the yellow dots in panel (b) of Figure 2. The yellow dots show a sinusoidal trend as expected from a position shift and are far away from the linear trend that we saw from a normal pulsar in the middle panel of Figure 1. But, if our new position is close to the true position, the sinusoidal trend should converge back to a linear line if we also correct the rotation period. The linear trend is different from the yellow dots only by an integral number of pulse phases. Here, we employ a small trick to help bring the residuals to roughly the correct rotation. The TOAs of the two most adjacent residual groups (marked as green crosses in the figure) are often close to phase-connected and reside in clearly the same rotation phase. We could use them to determine a likely correct linear trend that the TOAs should adhere to. Drawing a straight line across the two most adjacent points gives us the purple line in panel (b) of Figure 2. Then, we move the new residuals (yellow dots) to the same rotation as determined by the purple line by shifting them with an integral number of phases and marking them as green dots.

After we moved all the residuals to the respective rotation, then we removed the linear trend from them. In some cases,

for instance, in our test on PSR J2149+5643, the rotational-phase-corrected residuals, i.e., the green dots, did not all reside in one rotation phase. The corrected residual gradually grew to more than one and wrapped around the purple line. However, this behavior is very predictable, so we employed another step in our program that checks for a sudden jump in the residual phase and then corrects the phase wrapping by adding or subtracting one rotation from all the subsequent residuals, resulting in a nearly continuous final residual curve. If the assumed new position is correct, then the final residual should all be close to phase 0 after a linear trend is removed. We test if this is the case by calculating the χ^2 of the final residuals. In the third step, we identify the best-fit position that produced the smallest reduced χ^2 in the second step. The TOA residual produced by this new position, as calculated by the GPU program, is shown as the green dots in panel (c) of Figure 2. The corresponding $\Delta\alpha$, $\Delta\delta$, and new ν_0 are read out and used to update our timing ephemeris. In the fourth step, we apply the new ephemeris to the TOAs in PINT/TEMPO2, turn on fitting for the binary parameters, add eccentricity parameters if needed, and update the timing residuals. Since the correct position has been found, this newly updated ephemeris produced a flat residual (bottom plot of Figure 2).

It is worth mentioning that since this method of calculating the residuals is different from the timing software like TEMPO, TEMPO2, and PINT, the value of χ^2 may be slightly different, but the differences are negligible. As we can see in Figure 3, χ^2 is unacceptably high over the entire search region except in the few milliarseconds around the correct solution.

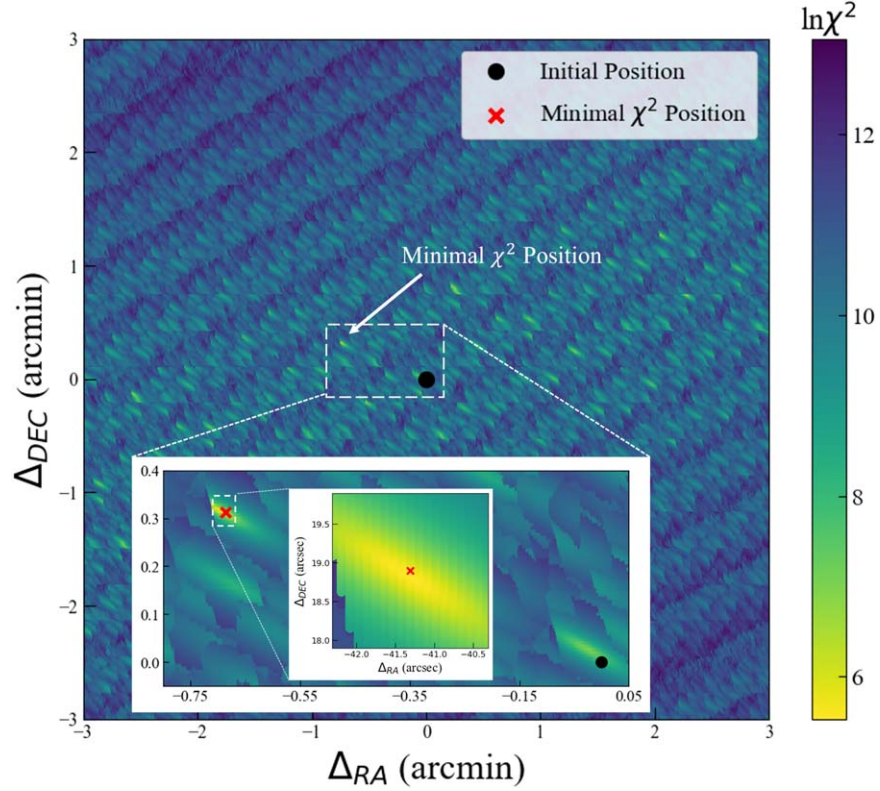


Figure 3. Distribution of residuals' reduced χ^2 over positional offsets.

3. Result

We tested our algorithm on solving two new pulsars discovered in the FAST CRAFTS survey⁹—an isolated pulsar PSR J0706+2707 and a binary pulsar PSR J2149+5643 from scratch. For the tests, we used an Intel Xeon Silver 4314 CPU and one NVIDIA A40 GPU to parallel compute 368,640,000 sets of solutions in each sub-grid using 4.4 GB memory. In testing pulsar J0706+2707, we used eight observations spanning one year to find the phase-connected timing solution in a quick search that took about two minutes. For the binary pulsar J2149+5643, we used a step of 1 mas and a range of 3' (both in R.A. and decl.). The search ultimately took 7.4 hr to complete.

PSR J0706+2707 is an isolated pulsar with a spin period of 70.3 ms. We find its phase-coherent solution easily in quick searching mode as illustrated in Figure 1. This is due to the proximity of the initial position to the correct position, and the deviation did not lead to a significant offset in pulse number.

Comparing the pulse number of the pre-fit and post-fit ephemeris that all pulses have been arranged in the correct global rotation count by the initial ephemeris, so the timing residuals could show a clear linear structure with which we can easily find the correct position. One should note that the same

pulsar could also be solved by using TEMPO/TEMPO2 through down-hill methods; our GPU program does not show any advantage in solving this problem. Nevertheless, this practice shows that the method functions as intended and leads to a correct solution.

PSR J2149+5643 is a millisecond pulsar in a binary system with a spin period of 15.3 ms and orbital period of 2.06 days; we discovered it in September 2021 and then observed it 17 times until 2023 December (Miao et al. 2024, in preparation). The timing residual of the initial model is shown in panel (a) of Figure 2. We commenced our GPU program when the time span of the observations reached one year. One can see that the last TOA falls far from the model prediction and lost phase solution (reduced $\chi^2 = 8344.083$). After we exhaustively tested all the R.A. and decl. $6'' \times 6''$ grid of 1 mas resolution, using the procedure described in Section 2, we found the lowest reduced $\chi^2 = 246.803$ residuals (green dots in panel (c) of Figure 2). After we add ν_0 and eccentricity relation parameter to the model and refit the TOAs using the GPU computed new position, the new best-fit residuals show a nice phase connection (panel (d) of Figure 2) (Miao et al. 2024 in preparation).

So far, we have demonstrated that our GPU program correctly found the position, period, and pulse number to phase connect all TOAs for an isolated pulsar and a millisecond

⁹ <http://crafts.bao.ac.cn/pulsar>

pulsar when solving them. A caveat is that the new parameters found by our GPU program do not include \dot{P} . Once we fit the TOAs with the new parameters plus a \dot{P} , we will get a new set of position parameters about 20 mas offset from the initial GPU solution. The new position parameters belong to the phase-coherent solution, which presents the most accurate position of existing observation data. This final solution also worked well connecting the TOAs from the following observations not considered in our test.

4. Conclusions

Our results show that the GPU program presented in this paper can easily solve an isolated pulsar and also help solve a binary system when conventional methods find it difficult.

Like in most previous binary pulsar-solving practices, to use our method, one needs to perform a dense orbital observation campaign first to get a good initial orbit model, such that the errors in the orbital model do not accumulate enough to throw our fitting off. One common source of error in the orbital model is the eccentricity of the orbit, which is often not easy to measure in the first few observations. Likely, for most MSP-white dwarf systems, the eccentricity is rather low, and the maximum offset caused by not including eccentricity is about ex/c , this is often smaller than a millisecond and does not accumulate over time. Inaccuracies in other orbital parameters, such as P_b could accumulate but also could be mitigated by making more than one TOA per epoch and making sure that the orbital parameters could get the pulsar periods of each observing epoch right first. Our GPU program only exhaustively searches in position space; for this to work, the binary solution must be sufficiently precise so as not to introduce large TOA errors in subsequent observations. Thus, planning the initial orbital campaign with enough dense observations is still important.

A caveat of our method is that the program does not yet search for spin period derivatives and, thus, does not work when the observation spans over years where the effect of the spin derivative is not negligible.

A second caveat of the method is that it tries to find a phase-connected solution that contains the correct pulse numbers but does not guarantee the solution is physically correct since the period derivative is omitted, and this causes the output position parameters to be offset from the true value, especially when the observing span is short. A physically trustworthy pulsar spin solution still takes over one year of phase-connected observations.

Nevertheless, we provide a new tool for solving difficult pulsars, and this work could enable further development of brute-force pulsar-solving tools.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (12041303), the National SKA Program of China (2020SKA0120200), the CAS Project for Young Scientists in Basic Research YSBR-063, the National Natural Science Foundation of China (NSFC grant Nos. 12203070 and Nos. 12203072), and the CAS-MPG LEGACY project. This work made use of the data from FAST (Five-hundred-meter Aperture Spherical Radio Telescope). FAST is a Chinese national mega-science facility operated by the National Astronomical Observatories, Chinese Academy of Sciences.

ORCID iDs

Weiwei Zhu (朱炜玮)  <https://orcid.org/0000-0001-5105-4058>
 Dejiang Zhou (周德江)  <https://orcid.org/0000-0002-6423-6106>
 Tianlu Chen (陈天禄)  <https://orcid.org/0000-0002-2944-2422>
 Jiarui Niu (牛佳瑞)  <https://orcid.org/0000-0001-8065-4191>
 Mengyao Xue (薛梦瑶)  <https://orcid.org/0000-0001-8018-1830>

References

- Alvarez-Muñiz, J., & Stanev, T. 2006, *JPhCS*, **47**, 126
- Antoniadis, J., Arzoumanian, Z., Babak, S., et al. 2022, *MNRAS*, **510**, 4873
- Antoniadis, J., Freire, P. C. C., Wex, N., et al. 2013, *Sci*, **340**, 448
- Berti, E., Barausse, E., Cardoso, V., et al. 2015, *CQGra*, **32**, 243001
- Edwards, R. T., Hobbs, G. B., & Manchester, R. N. 2006, *MNRAS*, **372**, 1549
- EPTA Collaboration, Antoniadis, J., Babak, S., et al. 2023, *A&A*, **678**, A48
- Fonseca, E., Pennucci, T. T., Ellis, J. A., et al. 2016, *ApJ*, **832**, 167
- Freire, P. C. C., & Ridolfi, A. 2018, *MNRAS*, **476**, 4794
- Freire, P. C. C., & Wex, N. 2024, *LRR*, **27**, 5
- Kramer, M., Stairs, I. H., Manchester, R. N., et al. 2021, *PhRvX*, **11**, 041050
- Lange, C., Camilo, F., Wex, N., et al. 2001, *MNRAS*, **326**, 274
- Lorimer, D. R., & Kramer, M. 2005, *Handbook of Pulsar Astronomy* (Cambridge: Cambridge Univ. Press)
- Luo, J., Ransom, S., Demorest, P., et al. 2021, *ApJ*, **911**, 45
- Manchester, R. N., Hobbs, G. B., Teoh, A., & Hobbs, M. 2005, *AJ*, **129**, 1993
- Nice, D., Demorest, P., Stairs, I., et al. 2015, *Tempo: Pulsar Timing Data Analysis*, Astrophysics Source Code Library, ascl:1509.002
- Phillips, C., & Ransom, S. 2022, *AJ*, **163**, 84
- Ransom, S. 2011, *PRESTO: Pulsar Exploration and Search Toolkit*, Astrophysics Source Code Library, ascl:1107.017
- Reardon, D. J., Zic, A., Shannon, R. M., et al. 2023, *ApJL*, **951**, L7
- Ruderman, M. A., & Sutherland, P. G. 1975, *ApJ*, **196**, 51
- Shapiro, I. I. 1964, *PhRvL*, **13**, 789
- Taylor, J. H., & Weisberg, J. M. 1989, *ApJ*, **345**, 434
- Taylor, J., Ransom, S., & Padmanab, P. V. 2024, *ApJ*, **964**, 128
- Wex, N. 2014, arXiv:1402.5594
- Xu, H., Chen, S., Guo, Y., et al. 2023, *RAA*, **23**, 075024